

Los resultados de todos los ejercicios incorporarán, además del código fuente, los comentarios precisos y necesarios para su fácil comprensión. No escatimes esfuerzos en comentar el código, es una buena práctica para aprender, además de ser muy útil para modificaciones o reutilizaciones futuras. Así como que el nombre de los objetos sea apropiado.

Añade documentos en formato Word con capturas de la salida por pantalla (al ejecutar la página) si consideras que queda más clara la resolución del ejercicio.

Deberás entregar dos versiones del resultado:

1. **Un único documento en Word sin comprimir:** que contenga todas las líneas de código, y por orden, primero el código **html**, luego el código **css** y por último el código **js**. Este archivo servirá para que el profesor haga anotaciones y/o correcciones a los alumnos.
2. **Los archivos de código fuente:** en un único archivo, ya sea de extensión: **html**, **js**, o **css**. Si precisas entregar varios archivos, comprímelos en un único **zip**. (No admito rar).

El nombre del archivo entregado comenzará por tu nombre y seguido por ExamenXX. Ejemplo: **FedericoExamen01zip**

Ejercicio Práctico 3º (3 puntos) se valora el ejercicio en conjunto.

Se trata de realizar una aplicación web para realizar algunas acciones en el DOM del navegador.

Se pide:

1. Crear una clase para generar elementos HTML del DOM de tal forma que:
 - a) Para crear un elemento solo se necesite indicar el **tag** del elemento.
 - b) Se genere automáticamente el valor para el atributo **id**, de manera que ese **valor** sea **“nombre_del_tag_” + número_correlativo_para_cada_tag**. Ejemplos de valores para id:
 - H1_003 (tercer párrafo)
 - P_001 (primer párrafo)
 - INPUT_007 (séptimo input)

Sugerencia: contar los elementos con ese tag que ya hay en el DOM

- c) El atributo id será un atributo de la clase y un atributo del elemento HTML
 - d) Los objetos de esta clase podrán almacenar su **“innerHTML”**, su **“innerText”** y todos los **nombres de las propiedades** que se hayan establecido en cada elemento para el atributo **“style”**.
 - e) Además de los métodos que consideres necesarios, habrá método para:
 - Agregar el elemento HTML como primer elemento hijo de otro elemento
 - Agregar el elemento HTML como último elemento hijo de otro elemento
 - Agregar como último elemento del body
 - Establecer nueva **propiedad** del atributo **style** para un elemento existente en el DOM
2. Crear una clase para trabajar con elementos de tipo texto (P, LI, H1, H2 ...) que herede de la anterior y aporte como funcionalidad el facilitar el cambio de color de texto.
 3. Crear una clase para trabajar con elementos con tag **“button”** que herede de la primera clase y aporte la funcionalidad que consideres oportuna.
 4. Utilizando el fichero index.html adjunto, que no podrás modificar, y con las clases anteriores, se pide:
 - a) Crear un **botón (tag button)** al inicio de body que:
 - Ocupe todo el ancho de la ventana. (Sugerencia: establece en **style** la propiedad **width**: 100%)
 - Al pasar el ratón por encima: que cambie el color de los elementos H1, volviendo al color original al salir.
 - Al hacer clic en este botón que cambie el color de los elementos H2 y se mantenga cambiado. Si se vuelve a hacer clic, que vuelvan al color original.

Fichero index3.html que se adjunta:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Introducción a JavaScript</title>
</head>
<body>

  <h1>JavaScript: El Lenguaje del Navegador Web</h1>

  <p>JavaScript es un lenguaje de programación de alto nivel que se utiliza principalmente para agregar interactividad a las páginas web.</p>

  <h2>Funcionalidades Principales</h2>

  <p>Algunas de las funcionalidades principales de JavaScript incluyen:</p>

  <ul>
    <li>Manipulación del DOM: Permite interactuar con los elementos de la página.</li>
    <li>Manejo de eventos: Responde a las acciones del usuario.</li>
    <li>Ajax: Facilita la comunicación asincrónica con el servidor.</li>
    <li>Programación orientada a objetos: Ofrece capacidades de programación orientada a objetos.</li>
  </ul>

  <h2>¿Por Qué JavaScript?</h2>

  <p>JavaScript es un componente fundamental para el desarrollo web moderno. Permite crear experiencias de usuario dinámicas y atractivas.</p>

  <h2>Frameworks y Bibliotecas</h2>

  <p>Además del JavaScript puro, existen numerosos frameworks y bibliotecas populares como Angular, React y Vue.js, que simplifican el desarrollo y mejoran la eficiencia.</p>

  <h1>JavaScript en la Práctica</h1>

  <p>En la práctica, JavaScript se utiliza para crear aplicaciones web interactivas, juegos en línea, efectos visuales y mucho más.</p>

  <script type="module" src="js/app.js"></script>
</body>
</html>
```

